## REMARKS

Claims 28-40 are pending in the application. Claims 28-40 stand rejected. In view of the following remarks, Applicant respectfully requests allowance of the application.

## CLAIMS DEFINE OVER THE PRIOR ART

### 102(e) Rejections

Claims 28, 29 and 31-40 stand rejected under 35 U.S.C. § 102(e) as allegedly being anticipated by U.S. Patent No 6,578,046 to Chang et al. Applicant traverses.

With regards to claims 28, 34, 36 and 38, claims 28, 34, 36 and 38 recite, in part:

> the iterator automatically:
> determining the format of the collection from the format definition;
> creating a reader object that can process collections of the determined format;
> associating the reader object with the collection;
> extracting the set of elements from the collection using the reader object;…

In contrast, Chang does not teach the above-mentioned iterator. For example, the Office Action cites Chang col. 8:5-23 as teaching "determining the format of the collection from the format definition." However, col. 8:5-23 merely lists Iterator 36 as a class and remarks that it contains subclasses, but does not teach Iterator 36 determining the format of a collection from a format definition, as in Applicant's claims.

The Office Action also cites Chang col. 8:41-60 and col. 10:1-13 as teaching "creating a reader object that can process collections of the determined format." However, Col. 8:41-60 is merely a list of Query Classes rather than the teaching of an iterator creating a reader object from a determined format, as in Applicant's claims. And, col. 10:1-13 merely refers to the creation of a query object to execute a query, which is not done by an iterator.

The Office Action further cites Chang col. 18:35-45 as teaching "associating the reader object with the collection." However, as discussed above, Chang does not teach an iterator creating a reader object and, hence, in col. 18:35-45, does not teach an iterator associating an iterator-created reader object with a collection. Rather, col. 18:35-45 merely discusses an interface Query 13, for a query object, which is not an iterator.

The Office Action also cites col. 7:57-col. 8:25 as teaching "extracting the set of elements from the collection using the reader object." However, as discussed above, Chang does not teach an iterator creating a reader object and, hence, in col. 7:57-col. 8:25, does not teach an iterator-created reader object extracting elements from a collection, as in Applicant's claims. Rather, col. 7:57-col. 8:25 merely discusses Query Evaluator Classes, which are not an iterator.

Furthermore, with regards to claim 28, claim 28 recites, in part:

> the iterator automatically:
>> …
>> instantiating a set of objects representing the set of
>> elements, wherein the set of objects can be iterated through and
>> operated on by the iterator; …

The Office Action cites Chang col. 21:3-30 as teaching "instantiating a set of objects representing the set of elements." However, Chang col. 21:3-30 does not teach an iterator performing the instantiating, as in Applicant's claim 28. Rather, col. 21:3-30 merely discusses a Federated Query creating an object, which is not an iterator.

With the regards to claim34, claim 34 recites, in part:

> the iterator automatically:
>> …
>> in response to a request to process an element in the set of
>> elements:
>> …
>> instantiating an object representing the element; …

The Office Action cites Chang col. 21:3-30 as teaching "instantiating an object representing the element." However as discussed above under claim 28, col. 21:3-30 does not teach an iterator performing instantiating.

With regards to Claim 36, claim 36 recites in part;

> the iterator automatically:
>> …
>> instantiating a set of objects representing the set of
>> elements … ; and
>> storing the set of objects on the storage for subsequence
>> use.

The Office Action cites Chang col. 21:3-30 as teaching "instantiating a set of objects representing the set of elements." However as discussed above, col. 21:3-30 does not teach an iterator performing instantiating.

The Office Action cites Chang col. 7:57-col. 8:2 as teaching "storing the set of objects on the storage for subsequence use." However as discussed above, col. 7:57-col. 8:2 does not teach an iterator performing storing.

With regards to claim 38, claim 38 recites, in part:

> the iterator automatically:
> ...
> in response to a request to process an element in the set of elements:
> ...
> instantiating an object representing the element;
> storing the object on the storage for subsequence
> use; ...

The Office Action cites Chang col. 21:3-30 as teaching "instantiating an object representing the element." However as discussed above, col. 21:3-30 does not teach an iterator performing instantiating.

The Office Action cites Chang col. 7:57-col. 8:2 as teaching "storing the object on the storage for subsequence use." However as discussed above, col. 7:57-col. 8:2 does not teach an iterator performing storing.

With regards to claims 39 and 40, claim 39 and 40 recite, in part:

> the iterator to automatically:
> determine the format of the collection from the format
> definition;
> create a reader object that can process collections of the
> determined format;
> associate the reader object with the collection;
> extract the set of elements from the collection using the
> reader object; ...

The Office Action cites Chang col. 8:5-23 as teaching "determine the format of the collection from the format definition." However, as discussed above, col. 8:5-23 merely lists

Iterator 36 as a class and merely remarks that it contains subclasses, but does not teach Iterator 36 determining the format of a collection from a format definition.

The Office Action also cites Chang col. 8:41-60 and col. 10:1-13 as teaching "create a reader object that can process collections of the determined format." However, as discussed above col. 8:41-60 is merely a list of Query Classes rather than the teaching of an iterator creating a reader object from a determined format, as in Applicant's claims. And, col. 10:1-13 merely refers to the creation of a query object to execute a query, which is not done by an iterator.

The Office Action further cites Chang col. 18:35-45 as teaching "associate the reader object with the collection." However, as discussed above, Chang does not teach an iterator creating a reader object and, hence, in col. 18:35-45, does not teach an iterator associating an iterator-created reader object with a collection. Rather, col. 18:35-45 merely discusses an interface Query 13, for a query object, which is not an iterator.

The Office Action also cites Chang col. 7:57-col. 8:25 as teaching "extract the set of elements from the collection using the reader object." However, as discussed above, Chang does not teach an iterator creating a reader object and, hence, in col. 7:57-col. 8:25, does not teach an iterator-created reader object extracting elements from a collection, as in Applicant's claims. Rather, col. 7:57-col. 8:25 merely discusses Query Evaluator Classes, which are not an iterator.

Furthermore with regards to claim 39, claim 39 recites in part:

the iterator to automatically:
...
instantiate a set of objects representing the set of elements,
...

The Office Action cites Chang col. 21:3-30 as teaching "instantiate a set of objects representing the set of elements." However as discussed above, col. 21:3-30 does not teach an iterator performing instantiating.

With regards to claim 40, claim 40 recites in part:

the iterator to automatically:
...
in response to a request to process an element in the set of elements:

…
instantiate an object representing the element;…

The Office Action cites col. 21:3-30 as teaching "instantiate an object representing the element." However as discussed above, col. 21:3-30 does not teach an iterator performing instantiating.

Thus, for at least the above reasons, claim 28, 34, 36, 38, 39 and 40 are not taught by Chang and the Applicant respectfully requests allowance of claims 28, 34, 36, 38, 39 and 40 and dependent claims 29-33, 35 and 37.

### 103(a) Rejection

Claim 30 stands rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over Chang in view of U.S. Patent Application Publication No. 2002/0073119 to Richard. Applicant traverses.

As a further feature on independent claim 28, the Applicant respectfully submits that claim 30 defines over Chang. The deficiency of Chang is not corrected by Richard because Richard also fails to teach the iterator of claim 28. Therefore, combining Chang with Richard would fail to provide the iterator of claim 28.

Therefore, claim 28 and its dependent claim 30 are patentable over the prior art. Applicant requests allowance of the claim 30.

## CONCLUSION

Applicant respectfully submits that this application is in condition for allowance. A Notice of Allowance is earnestly solicited.

The Examiner is invited to contact the undersigned at (202) 220-4255 to discuss any matter concerning this application. The Office is hereby authorized to charge any additional fees or credit any overpayments under 37 C.F.R. § 1.16 or § 1.17 to Deposit Account No. 11-0600.

Respectfully submitted,

KENYON & KENYON LLP

Dated: <u>March 19, 2007</u>

/Cassandra T. Swain, Ph.D./
Cassandra T. Swain
Reg. No. 48,361

1500 K Street, NW
Suite 700
Washington, DC 20005
(202) 220-4200 telephone
(202) 220-4201 facsimile